様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　５ 年　 ６ 月 ２６ 日現在

機関番号：３７１１１
研究種目：基盤研究(C)（一般）
研究期間：2017～2022
課題番号：１７Ｋ００１１６

研究課題名（和文）自然言語処理技術による既存開発文書からの追加的要求定義に関する研究

研究課題名（英文）Elicitation of Additional Requirements from Existing Development Artifacts by Natural Language Processing

研究代表者

中西　恒夫（Nakanishi, Tsuneo）

福岡大学・工学部・教授

研究者番号：７０３１１７８５

交付決定額（研究期間全体）：（直接経費）　　3,500,000円

研究成果の概要（和文）：今日のソフトウェア開発現場では既存製品を改修して派生製品を開発することが主である。また，依然，開発現場の「第一言語」は自然言語である。そこで自然言語処理技術を用いて，既存製品の要求・仕様記述から，派生製品の考え得る要求・仕様を導出し，派生開発やプロダクトライン開発における開発者の負担を軽減する方法論に係る研究を実施した。製品間の共通性／可変性を表現するフィーチャモデルを半自動的に抽出する手法，事物語に修飾語を選択・適用し派生製品の要求獲得を支援する手法，これら手法のために並列構造を含む文を単文に分解する前処理手法を確立するとともに，形態素レベルでのパターンマッチングを行うツール群を開発した。

研究成果の学術的意義や社会的意義
プロダクトライン開発の現場において，自然言語で書かれた既存ソフトウェアの開発文書からフィーチャモデルをいかに生成するかは工数上，大きな問題となる。既存手法は統計的な手段に頼るものが多いが，広く使われている係り受け解析ツールを用い，解析的手段を用いる本研究の手法は新規なものである。提案手法が有効に機能するためには前処理が必要であるが，本研究ではその手法も確立した。一連の形態素レベルのパターンマッチングツールは，研究の副産物であるが，気の利いたツールとして産業上の実用性が高いものである。類似のツールはあるが日本語対応しておらず，さらに本ツールは係り受けのパターンマッチングにも使用できる。

研究成果の概要（英文）：Common software development projects usually perform development of derivative products by modifying existing products. The development projects use natural languages as the "first language" for documentation in many cases. Therefore, we conducted a study on a methodology to derive possible requirements and specifications of derived products from requirements and specifications of existing products using natural language processing technologies to reduce developers' workload in derivative development and product line development. We established a method for semi-automatically producing feature models that represent commonality/variability among products, a method for helping requirements acquisition for derived products by selecting and applying modifiers to object words, and a preprocessing method for decomposing sentences containing parallel structures into single sentences for these methods. Moreover, we also developed a set of tools for pattern matching at the morphological level.

研究分野：ソフトウェア工学

キーワード：ソフトウェア要求　自然言語処理　プロダクトライン開発　派生開発　フィーチャモデル　単文化　パターンマッチング

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９（共通）

１．研究開始当初の背景

自然言語の記述が持つ曖昧さと自由さは，開発するシステムやソフトウェアへの欠陥の混入を招くうえに開発の自動化を困難にする。そのため，ソフトウェア工学分野では，形式手法，モデル駆動開発，モデルベース開発等，ソフトウェア開発における自然言語の仕様を制限する方向の力学が絶えず働いてきた。しかしながら，現実のソフトウェア開発現場を見れば，依然，「第一言語」は自然言語のままであり，この状況は研究開始当初も現在も変わっていない。

比較的高い抽象度と形式性を有するモデルの導入が，開発現場において重要視されてはいても進まない，あるいは部分的である理由としては，i) モデル導入による品質・コスト・工期への改善効果が容易には読めないこと，ii) モデルの記述が短期的には純粋なコスト増となること，iii) 導入に係る教育等のコストや現場の負担感や抵抗感といった導入障壁があること，iv) 自然言語で書かれた既存資産からモデルを起こすには工数を要すること，v) 既存モデリング言語の記述力に満足できないこと等が挙げられる。こうしたソフトウェア開発現場の現実を見れば，システム／ソフトウェア開発に係る文書がすべてモデルに置き換わることは考えがたい状態であった。この状況もまた変わっていない。

一方で，十分実用に耐える精度の自然言語処理ツールが広く利用可能となっており，100%の自動化は無理にしても，システム／ソフトウェア開発の QCD を改善し，システム／ソフトウェア開発者の負担を下げるために，そこそこの自動化を行える環境はすでに整っている状態となっていた。

２．研究の目的

本研究では，システム／ソフトウェア開発において自然言語による記述は比率こそ変わるものの残り続けるとの立場をとり，開発プロセス中，特にモレヌケ，曖昧さ，矛盾に係る欠陥が作り込まれがちな要求定義工程に自然言語処理技術を導入することで，自然言語で記述された製品の開発文書から，マン・マシン協調的，かつインクリメンタルに要求を獲得する手法の確立を図った。

今日のシステム／ソフトウェア開発では，スクラッチからの開発をすることは稀であり，既存製品の強化や変更に係る開発が主であることから，本研究では，派生開発，あるいはプロダクトライン開発を支援すべく，既存製品の要求・仕様記述文書から派生製品の要求・仕様の導出を図ることとした。

３．研究の方法

本研究以前に研究代表者が実施していた，HAZOP をソフトウェアの異常処理要求の獲得に用いる研究（「ソフトウェア FMEA の一手法とプロダクトライン開発におけるその利用」，信学技報，Vol.111，No.481，2012 年 3 月）に自然言語処理を導入する方向で進めた。同研究では，自然言語によるシステム／ソフトウェアの振舞い要求記述から，HAZOP のガイドワードを用いてソフトウェアの振舞いにおける正常からの逸脱状態，すなわち異常状態を想起し，異常処理のための要求を人手で獲得するプロセスを提唱していた。

本研究では，上述のプロセスに自然言語処理における述語項構造解析の手法を導入し，自然言語で記述された既存製品の開発文書から派生製品の要求を抽出する手法の確立を図ることとした。

４．研究成果

本研究で得られた成果を以下に述べる。

(1) 要求・仕様記述文の共通性／可変性分析と派生的導出： 自然言語で記述された開発文書から派生開発，あるいはプロダクトライン開発における機能／非機能の要求・仕様記述文の候補を半自動的に導出する手法を開発した（中西，2019）。

この手法では，製品群の一連の要求・仕様記述文に対して，文節間の係り受け構造を表現する木構造を生成し，それらの木構造をマージすることで，文の間の共通部分と相違部分を分離した木構造にする。文節間の係り受け構造は cabocha や knp 等，既存の自然言語処理ツールで生成できる。 得られた木構造をもとに，プロダクトライン分野で製品間の共通性／相違性を表現するモデルとしてデファクトスタンダード的に用いられているフィーチャモデルを構築する。よいフィーチャ名を与える必要があるため，この工程は人手に頼る必要がある。さらに，あらかじめ用意される概念辞書（いわゆるオントロジ）と書き換え規則に基づいて，原要求・仕様記述文から派生的要求・仕様記述文を生成する。

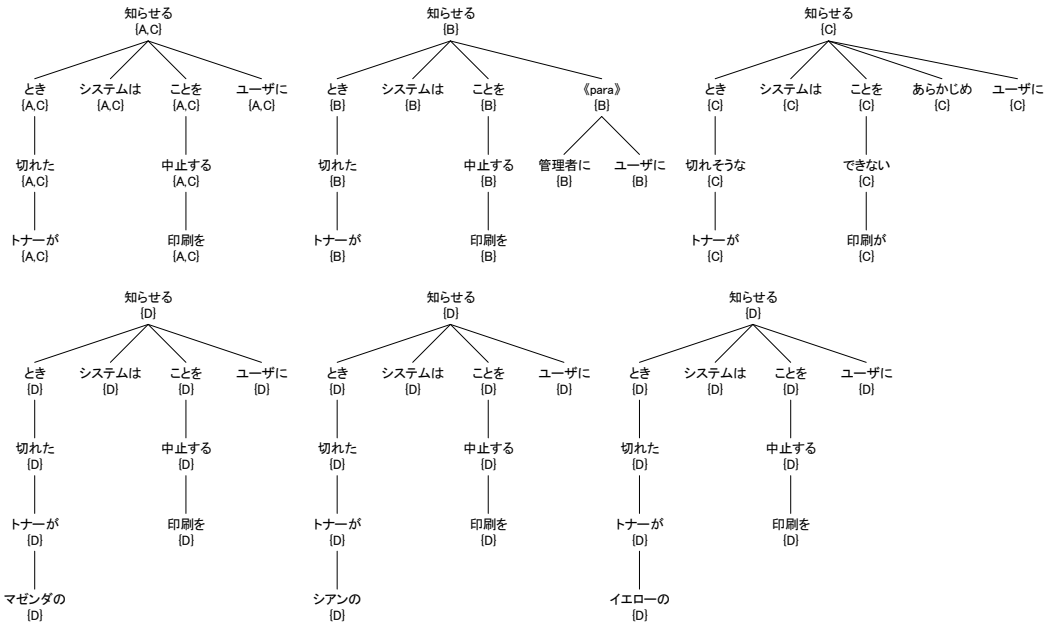
知らせる
{A,C}
とき
{A,C}
システムは
{A,C}
ことを
{A,C}
ユーザに
{A,C}
切れた
{A,C}
中止する
{A,C}
トナーが
{A,C}
印刷を
{A,C}
知らせる
{B}
とき
{B}
システムは
{B}
ことを
{B}
《para》
{B}
切れた
{B}
中止する
{B}
管理者に
{B}
ユーザに
{B}
トナーが
{B}
印刷を
{B}
知らせる
{C}
とき
{C}
システムは
{C}
ことを
{C}
あらかじめ
{C}
ユーザに
{C}
切れそうな
{C}
できない
{C}
トナーが
{C}
印刷が
{C}
知らせる
{D}
とき
{D}
システムは
{D}
ことを
{D}
ユーザに
{D}
切れた
{D}
中止する
{D}
トナーが
{D}
印刷を
{D}
マゼンダの
{D}
知らせる
{D}
とき
{D}
システムは
{D}
ことを
{D}
ユーザに
{D}
切れた
{D}
中止する
{D}
トナーが
{D}
印刷を
{D}
シアンの
{D}
知らせる
{D}
とき
{D}
システムは
{D}
ことを
{D}
ユーザに
{D}
切れた
{D}
中止する
{D}
トナーが
{D}
印刷を
{D}
イエローの
{D}

図 1

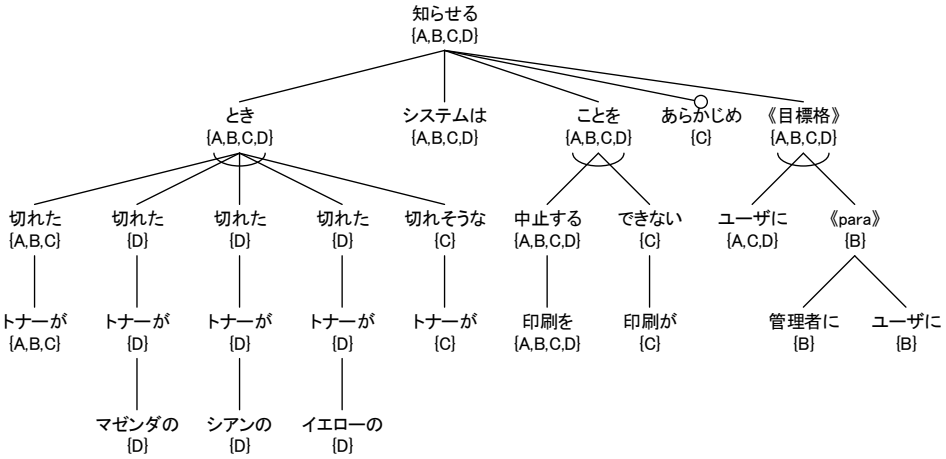
知らせる
{A,B,C,D}
とき
{A,B,C,D}
システムは
{A,B,C,D}
ことを
{A,B,C,D}
あらかじめ
{C}
《目標格》
{A,B,C,D}
切れた
{A,B,C}
切れた
{D}
切れた
{D}
切れた
{D}
切れそうな
{C}
中止する
{A,B,C,D}
できない
{C}
ユーザに
{A,C,D}
《para》
{B}
トナーが
{A,B,C}
トナーが
{D}
トナーが
{D}
トナーが
{D}
トナーが
{C}
印刷を
{A,B,C,D}
印刷が
{C}
管理者に
{B}
ユーザに
{B}
マゼンダの
{D}
シアンの
{D}
イエローの
{D}

図 2

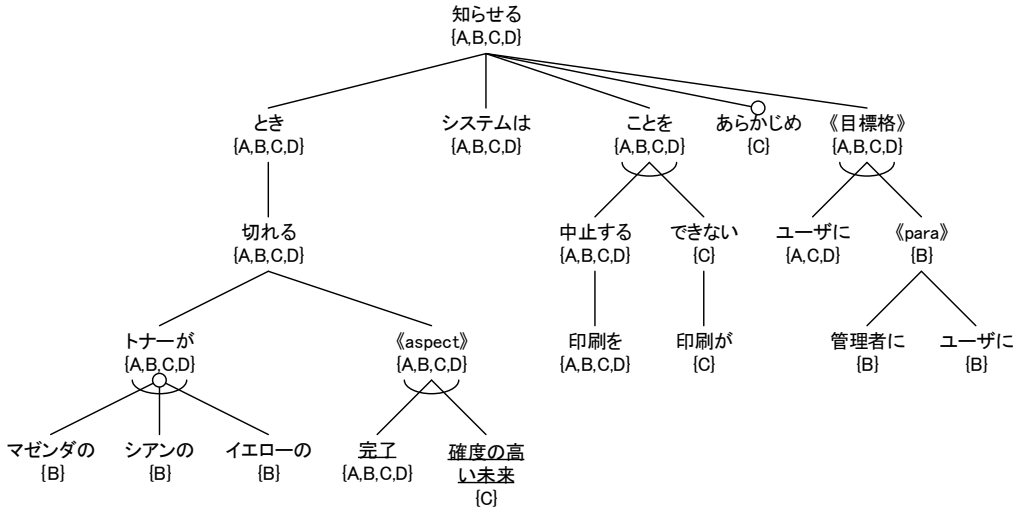
知らせる
{A,B,C,D}
とき
{A,B,C,D}
システムは
{A,B,C,D}
ことを
{A,B,C,D}
あらかじめ
{C}
《目標格》
{A,B,C,D}
切れる
{A,B,C,D}
中止する
{A,B,C,D}
できない
{C}
ユーザに
{A,C,D}
《para》
{B}
トナーが
{A,B,C,D}
《aspect》
{A,B,C,D}
印刷を
{A,B,C,D}
印刷が
{C}
管理者に
{B}
ユーザに
{B}
マゼンダの
{B}
シアンの
{B}
イエローの
{B}
完了
{A,B,C,D}
確度の高い未来
{C}

図 3

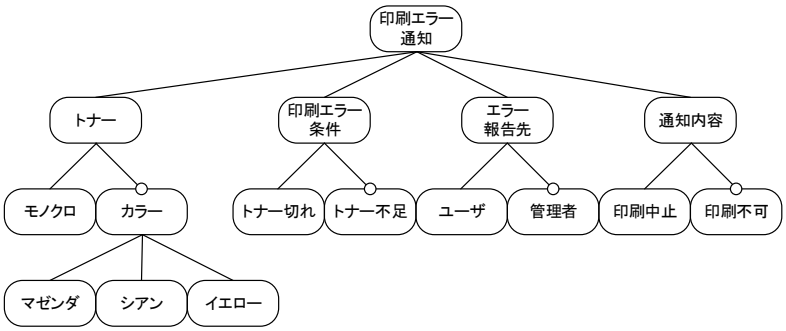

図 4

提案手法の適用例を示す。図 1 は，「トナーがきれたときシステムは印刷を中止することをユーザに知らせる」，「トナーが切れたときシステムは印刷を中止することをユーザに知らせる」，他 4 件の要求記述文の係り受け構造を表現する木構造である。これらの木をマージしたものが図 2 であり，以下再帰的にマージを繰り返し，最終的に図 3 のかたちになる。最後に人手に依る解釈を経て図 4 のようなフィーチャモデルを得る。

自然言語で書かれた要求・仕様記述文から自動的にフィーチャを抽出する研究は多数行われているが，統計的手段によるものが多く，本手法のように解析的手段によるものは見られない。一方で，統計的手段による手法と異なり，提案手法は要求・仕様記述文が製品間でおおよそ揃った係り受け構造になっていなければ有効に機能しない可能性がある。(派生製品を開発する際に，よく似た既存製品の要求仕様書を丸々再利用し，追加・変更のあった部分のみ改変をしてきた開発現場ではそのようになっている可能性は高い。) 提案手法が有効に機能するようにするため，複雑な要求・仕様文を前処理し，単純な構造の文に変換しておく必要がある。

(2) ガイドワードを用いた要求・仕様記述文の派生的導出プロセス： 派生製品の，特に安全性や信頼性に関する要求を抽出するプロセスとして，フリーガイドワード HAZOP を提案し，簡単なケーススタディを用いた評価を行った。システムの安全性，信頼性を分析する古典的な手法として知られる HAZOP は，システムが取り扱うさまざまな物理量や特性量といった「量」の正常範囲からの逸脱を異常と捉え，その対策や予防を検討する方法論である。HAZOP では「過大な」「過小の」「ゼロの」「逆の」「異なる」といった限られたガイドワードでこうした「量」を修飾することで異常な状況を想起するということが行われる。しかしながら，システムの具体的なかたちが定まっていない上流工程において，要求・仕様記述文のような抽象度の高い記述から，そうした「量」を見出し，ガイドワードで修飾した状況を解釈することは容易ではない。そこで本研究では，限られたガイドワードで「量」を修飾するのではなく，(既存製品の要求・仕様記述文に現れる事物語を適当な形容詞句で修飾することで異常な状況を想起するアプローチを採った。たとえば，従来の HAZOP では，タンクに汚れた水を入れる状況は，従来の HAZOP なら要求・仕様記述文に現れる「水」という事物語に対し，「清浄度」という量を想起し，それに「過小の」というガイドワードをかけることでようやく想起できるものであったが，フリーガイドワード HAZOP では，「水」に「汚い」という修飾語をかけることで直接的に想起可能となる。

形容詞句は無数に存在するため，対象の事物語を修飾し得るものであり，かつ派生製品の安全性や信頼性に関する要求・仕様を抽出するうえで効果的なものを選択する必要がある。当初，辞書（分類語彙表）を用いて，語彙カテゴリに基づいて修飾語（形容詞）を絞るアプローチを採ったが（中西，2018），事物語にあり得ない修飾語がかけられるケースがほとんどであった。その後，Wikipedia の記事に係り受け解析を適用し，事物語を実際に修飾する語を登録した辞書を生成，使用するよう方法を改めた。また，実際にトマト収穫ロボットの異常処理に関する追加／変更要求仕様を試みるケーススタディを実施した（中西，2022)。結果，あり得ない修飾語がかけられるケースはなくなったものの，修飾語の数は多く，異常状態の想起に効果的なものを選別することは十分にできておらず，手法適用の負担感は否めない結果となった。

異常状態の想起に効果的な修飾語を提供する手段の確立が依然課題となっており，そのためにより適切なコーパスを収集，選択する方策を検討してきたが，今後は大規模言語モデルを活用することがより現実的な解であると考えている。

(3) 要求・仕様記述文の単文化： 上述の要求・仕様記述文の共通性／可変性解析，ならびにフリーガイドワード HAZOP においても，元となる製品群の要求・仕様記述文の前処理，特に複雑な要求・仕様文を分解し，定型的な単純な構造の文に分解し，機械的な処理が容易にしておくことが必要となる。自然言語による技術文書の場合，冗長な表現を避けるため，並列構造を有する文が多用されるが，既存の係り受け解析器は並列構造を有する文の解析を正しく行えないことが多いため，本研究では，単一化文法を用い，並列構造を有する要求・仕様記述文を単文化することに取り組んだ。

単一化文法は，文脈自由文法（CFG）における非終端記号を単なる記号ではなく，素性とその

値の対で構成される素性構造で表現し，CFG における書き換え規則の適用条件を素性構造の単一化で表現する文法である。非終端記号，ひいては句に関するさまざまな意味情報を素性に持たせることで，句の性格や意味に深く立ち入った多様な構文解析を統一的な枠組みとプロセスで実施できる（吉村，2018）。

日本語文の代表的な並列構造は，「コーヒー豆と水」のような名詞並列，「水を入れて，スイッチを押す。」のような述語並列，「フィルタにコーヒー豆，タンクに水を入れる。」のような句の部分的構造を並列化する部分並列がある。名詞並列，述語並列については単一化文法の枠組みで構文規則を記述でき，入力文の解析結果に基づいて入力文を構成する単文に分解できるが，部分並列については単一化文法を用いた構文規則を記述することが容易ではなかった。そのため，部分並列を含む入力文については，解析が失敗した時点（入力文全体の構造が得られていないにもかかわらず書き換え規則の適用がそれ以上できなくなった時点）までの解析結果を用いて，入力文を単文に分解するアルゴリズムを開発した（中村，2019；内野，2022）。

しかしながら，このアルゴリズムは単文への分割はできるものの，それまでの解析情報を一旦捨てて，得られた単文それぞれについて構文解析を一からやり直す必要があった。そこで，解析が失敗して部分並列構造を検出した時点で，部分並列構造をそれまでの解析情報に新たに追加し，複数の単文の構造解析を並列に行うよう手法を改良した（北川，2023）。

以上の一連のアルゴリズムにより，基本的な名詞並列，述語並列，部分並列を含む入力文を単文に分解できるようになったが，現時点では「コーヒー豆と水をそれぞれフィルタとタンクに入れる。」のような，特殊な副詞「それぞれ」を使った並列構造を含む文は単文に分解することができていない。こうした特殊な副詞を使った並列構造を有する入力文を単文に分解できるようにすることが課題として残っている。

(4) 形態素レベルパターンマッチャの開発： 本研究では，日本語文に対するルールベースの処理を行う必要性から，自然言語文の形態素レベルでのパターンマッチングを行う形態素レベルパターンマッチャ morfgrep（中西，2019），ならびにその派生ソフトウェア群 morfawk（中西，2020），morfawk.ja（中西，2023）が副産物として開発された。UNIX のツール，grep が正規表現で指定された文字レベルのパターンマッチングを行うのに対して，morfgrep は正規表現で指定された形態素レベルのパターンマッチングを行う。たとえば，名詞の 1 個以上の連続，つまり複合名詞を検索したい場合は正規表現「¥名詞+」を指定する。形態素のパターンは，基本的に品詞をもとに指定するが，品詞の下位クラス（たとえば「名詞」に対して「固有名詞」）を指定したり，あるいはマッチする形態素に対する表層形，原形，活用型といった各種属性に対する制約を指定したりすることも可能である。

morfgrep がパターンマッチした部分を出力するだけであるのに対して，UNIX のツール，awk が文字レベルのパターンに応じて実行すべき処理を記述できるように，morfawk は形態素レベルのパターンに応じて実行すべき処理を Python で記述できる。

さらに，morfawk.ja は文節単位の係り受け構造に対してパターンマッチングを行い，マッチしたパターンに応じて実行すべき処理を記述することが可能である。修飾する文節，ならびに修飾される文節を構成する形態素列のパターンを morfgrep 同様に指定でき，さらに修飾関係に関するパターンを指定できる。morfawk 同様，マッチしたパターンに応じて実行すべき処理を Python で記述する。以下に，「〇〇信号を～から･･･へ転送{さ，し，しろ，する，すれ，せ}」にマッチする部分を検索し表示する，morfgrep.ja のスクリプトを示す。文節の出現順序ではなく，係り受け関係でマッチをするため，文節の順番がちがっていてもマッチすることに注意されたい。

```
(¥名詞+?¥名詞<表層形=="信号">)@Sig¥助詞<表層形=="を">
    =>(¥名詞.サ変<表層形=="転送">¥動詞<原形=="する">)@V,
¥名詞@Src¥助詞<表層形=="から"> => ¥@V,
¥名詞+@Sink¥助詞<表層形=="へ"> => ¥@V:
    print("Signal = ", NML["Sig"])
    print("Source = ", NML["Src"])
    print("Sink = ", NML["Sink"])
```

後に，morfgrep とほぼ同様のツールが Stanford Natural Language Processing Group より TokensRegex として開発されていることが明らかになった。morfgrep は日本語を対象とできること以外に差異点を見出せていないが，morfawk，morfawk.ja についてはマッチしたパターンに応じて実行すべき処理を記述できる点，係り受け構造に対してパターンマッチングができる点が差異点であると認識している。

５．主な発表論文等

〔雑誌論文〕　計1件（うち査読付論文　1件／うち国際共著　0件／うちオープンアクセス　1件）

１．著者名 田辺 利文，乙武 北斗，吉村 賢治，中西 恒夫，古庄 裕貴	４．巻 100
２．論文標題 ソフトウェア開発プロセスにおける自然言語処理ツールの活用	５．発行年 2017年
３．雑誌名 福岡大学工学集報	６．最初と最後の頁 81-85
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

〔学会発表〕　計11件（うち招待講演　2件／うち国際学会　1件）

１．発表者名 内野 皓介，田辺 利文，乙武 北斗，吉村 賢治
２．発表標題 日本語単一化文法における並列構造の解析
３．学会等名 情報処理学会火の国シンポジウム
４．発表年 2022年

１．発表者名 中西 恒夫，乙武 北斗，藤永 拓矢，吉村 賢治，田辺 利文
２．発表標題 フリーガイドワードHAZOPのための係り受け解析を用いた修飾語辞書の生成と利用
３．学会等名 情報処理学会：知能ソフトウェア工学研究会
４．発表年 2022年

１．発表者名 中西 恒夫，吉村 賢治，乙武 北斗，田辺 利文，古庄 裕貴，西浦 洋一
２．発表標題 morfawk：形態素パターンマッチング／処理言語
３．学会等名 電子情報通信学会：ソフトウェアサイエンス研究会
４．発表年 2020年

１．発表者名
中西 恒夫，吉村 賢治，乙武 北斗，田辺 利文，古庄 裕貴，西浦 洋一，浅野 雅樹

２．発表標題
形態素パターンマッチャmorfgrep とそのソ フトウェア開発における応用

３．学会等名
電子情報通信学会：ソフトウェアサイエンス研究会

４．発表年
2019年

１．発表者名
中西 恒夫

２．発表標題
プロダクトライン開発におけるNLP応用

３．学会等名
システム開発文書品質研究会：ASDoQ大会2019（招待講演）

４．発表年
2019年

１．発表者名
乙武 北斗

２．発表標題
ソフトウェア技術者のための自然言語処理技術

３．学会等名
システム開発文書品質研究会：ASDoQ大会2019（招待講演）

４．発表年
2019年

１．発表者名
中西 恒夫，吉村 賢治，乙武 北斗，田辺 利文，古庄 裕貴

２．発表標題
要求・仕様記述文の共通性／可変性分析と派生的導出に関する一手法

３．学会等名
情報処理学会：ソフトウェア工学研究会

４．発表年
2019年

１．発表者名
中村 健，乙武 北斗，田辺 利文，吉村 賢治

２．発表標題
単一化文法を用いた日本語文の構文解析における並列構造の処理

３．学会等名
情報処理学会第81回全国大会

４．発表年
2019年

１．発表者名
中西 恒夫，吉村 賢治，乙武 北斗，田辺 利文，古庄 裕貴

２．発表標題
フリーガイドワードHAZOPプロセスにおける分類語彙表データベースの活用

３．学会等名
電子情報通信学会：知能ソフトウェア工学研究会

４．発表年
2018年

１．発表者名
吉村 賢治，中西 恒夫，乙武 北斗，田辺 利文，古庄 裕貴

２．発表標題
日本語単一化文法を用いた開発文書からの単文抽出

３．学会等名
電子情報通信学会：知能ソフトウェア工学研究会

４．発表年
2018年

１．発表者名
Tsuneo Nakanishi, Hokuto Ototake, Toshifumi Tanabe, Kenji Yoshimura

２．発表標題
Morfawk.ja: A Japanese Token Level Pattern Matching and Processing Language with Dependency Analysis

３．学会等名
Int. Conf. on Software and Computer Applications (ICSCA) 2023（国際学会）

４．発表年
2023年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

6．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考
研究分担者	吉村　賢治 (Yoshimura Kenji) (40167002)	福岡大学・工学部・教授 (37111)	
研究分担者	乙武　北斗 (Ototake Hokuto) (20580179)	福岡大学・工学部・助教 (37111)	
研究分担者	古庄　裕貴 (Furusho Hiroki) (90781807)	福岡大学・工学部・助教 (37111)	
研究分担者	田辺　利文 (Tanabe Toshifumi) (80330900)	福岡大学・工学部・助教 (37111)	
研究分担者	廣重　法道 (Hirosige Norimichi) (30736228)	福岡大学・工学部・助教 (37111)	

7．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

8．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関